BUSINESS TRANSFORMATION

INSIGHTSTUDIO

From IT delivery queue to a business-driven apps marketplace.

Sales ops, customer success, finance, people ops, and product ops compose AI-powered internal apps directly — on a governed platform. The long tail of departmental apps moves off the IT backlog. IT runs the fabric beneath every app.

THE SHIFT

BEFORE

Business asks. IT builds. Eventually.

- 14-month IT backlog of business-app requests
- Shadow IT fills the gap — Power Automate, Zapier, browser-tab ChatGPT
- Customer data in unsanctioned tools; each app a one-off; no SSO, no audit

AFTER · WITH INSIGHTSTUDIO

Business composes. IT enables.

- Apps composed in **a week**, not a quarter
- Internal marketplace — Okta-style access requests
- Every app inherits SSO, audit, secrets, connectors to Salesforce, SAP, Snowflake, SharePoint, Outlook

THE APPROACH

ALL THE AVAILABLE AI TOOLS — AND WHERE INSIGHTSTUDIO FITS

ChatGPT / Claude · M365 Copilot · Salesforce Einstein · Power Platform + Copilot Studio · ServiceNow App Engine · Salesforce Agentforce · Lovable / v0 / Bolt · **InsightStudio**

Chat assistants and copilots boost individual productivity inside existing tools. Vibe-coding builders (Lovable, v0, Bolt) are great for prototypes and design exploration. Vendor platforms (ServiceNow, Salesforce, Microsoft) work well inside those ecosystems. **InsightStudio is the pick when you want a self-hosted, open, vendor-neutral fabric where business teams compose governed apps that integrate with every system, not just one vendor's.**

CUSTOMER PATTERN

TECHNOLOGY · MID-SIZE B2B SAAS

Internal apps marketplace for non-engineering teams

WHAT RUNS ON THE MARKETPLACE TODAY

Sales ops, customer success, finance, people ops, and product ops compose directly. GitHub Copilot remained in engineering's IDE; InsightStudio absorbed the long tail.

OUTCOMES AFTER 12 MONTHS

14 → 4	0 → 63	5	$3.2M → $1.4M
MONTHS · IT BACKLOG	APPS IN PRODUCTION	BUSINESS TEAMS ONBOARDED	ANNUAL AI SPEND

Bring departmental apps off the IT backlog and into the marketplace.

90-day pilot: stand up InsightStudio in a non-production VPC, train 2–3 business builders, ship their first governed apps in under two weeks. Reversible at every phase.

REQUEST PILOT →

DEVELOPER TRANSFORMATION

INSIGHTWORKER · SDD

Write the spec once. Agents derive the code, the tests, and the docs.

Engineering organizations that adopt spec-driven development with InsightWorker stop maintaining three artifacts in parallel. The spec becomes the source of truth. When it changes, the agent re-derives the implementation, regenerates the tests, updates the documentation — and opens a pull request with the diff.

THE SHIFT

BEFORE

Three artifacts drift apart over time.

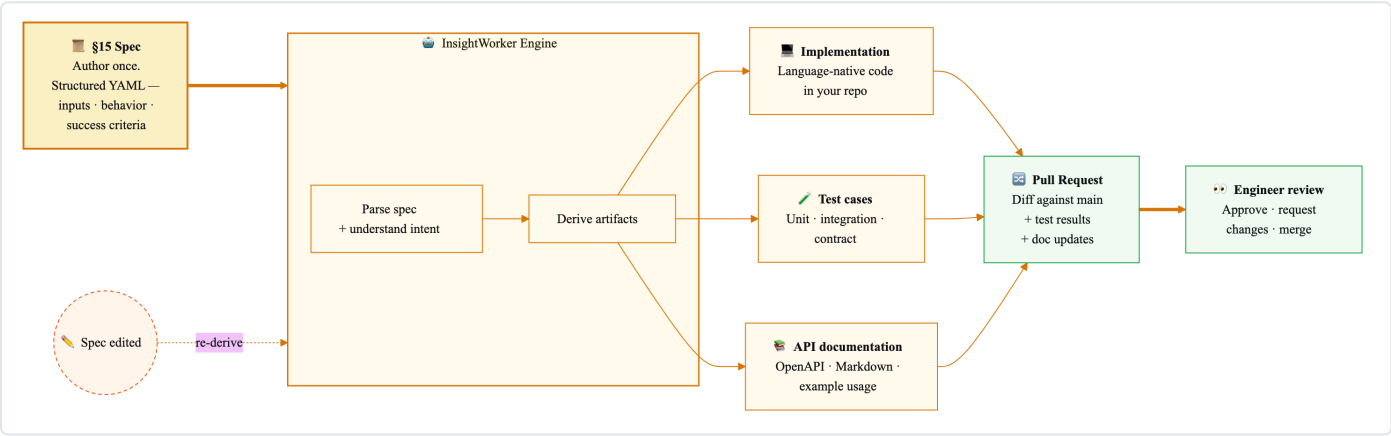
- Spec in Confluence; code in repo; tests somewhere else
- Engineer reads spec, writes code + tests + docs by hand
- Spec changes; code updated; tests + docs fall behind
- Six months later, three artifacts tell three different stories

AFTER · WITH INSIGHTWORKER SDD

One spec. Three derived artifacts. Always in sync.

- Spec authored once in the structured §15 format
- Agents generate implementation + test cases + API documentation
- Spec change → agent re-derives diff and opens a PR
- Repo always reflects the current spec; engineers review, don't author

THE APPROACH



ALL THE AVAILABLE AI TOOLS — AND WHERE INSIGHTWORKER SDD FITS

GitHub Copilot (IDE) · Cursor · Tabnine · Claude Code (agentic) · Devin · Aider · Lovable / v0 / Bolt (prototypes)

InsightWorker SDD

Keep Copilot in the IDE for code completion. Keep Claude Code and Devin for repo-wide refactors. Keep Lovable / v0 / Bolt for prototype exploration. **InsightWorker SDD is the layer that ties them together — making the spec the authoritative artifact and keeping implementation, tests, and docs honest to it. It coexists with, rather than replaces, your existing developer-AI tooling.**

CUSTOMER PATTERN

RETAIL · NATIONAL ENTERPRISE

Spec-driven development across the engineering org

WHAT INSIGHTWORKER DOES FOR THEM

Specs authored once in §15. Agents derive the implementation, test cases, API documentation in lockstep — and re-derive when the spec changes.

OUTCOMES

~60%	100%	Zero	Hours
FASTER · NEW ENDPOINT CYCLE	API DOCS · AUTO-SYNC'D	MANUAL SCAFFOLDING FROM SPEC	VS DAYS · SPEC-CHANGE PR

Make the spec the source of truth — keep code, tests, and docs honest to it.

Pilot SDD on a single service or repo. Author the next planned spec change in §15 format; let the agent derive the diff. The InsightWorker engine is open — every bundle is YAML you can read.

REQUEST PILOT →

QA TRANSFORMATION

INSIGHTTESTBENCH

Product owners and test leads drive automated testing directly.

No more Playwright authoring by hand. Paste a brief; the bench logs into your app via SSO, crawls every page, builds a feature tree, and generates regression / responsive / UX-audit plans grounded in real screenshots and DOM. Failures diagnose themselves. Confirmed product bugs become pull requests.

THE SHIFT

BEFORE

Regression is a fortnight of QA effort.

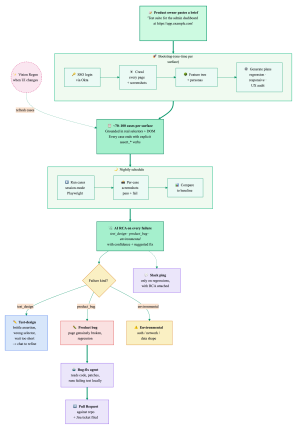
- QA engineers writing Playwright by hand, per case
- 30% of CI failures dismissed as "probably the test"
- UI changed? Rewrite weeks of cases
- Half the suite has accumulated `@skip` markers

AFTER · WITH INSIGHTTESTBENCH

Regression is a scheduled overnight job.

- Product owner pastes a brief; bench generates ~70-100 cases
- Cases grounded in real screenshots + DOM — no invented selectors
- UI changed? Vision Regen refreshes cases from the new page
- Nightly run + baseline compare + AI RCA on every failure; Slack only on real regressions

THE APPROACH



ALL THE AVAILABLE AI TOOLS — AND WHERE INSIGHTTESTBENCH FITS

Manual Playwright | Cypress Studio | Selenium IDE | Mabl / Testim (vendor SaaS) | QA-Copilot in Cursor | Lovable / v0 (UI prototypes only) | **InsightTestBench**

Existing tools help engineers write tests faster. Vendor SaaS test platforms work, but ship your test data outside your environment. **InsightTestBench is the pick when you want product owners and test leads to drive QA without writing Playwright — and you want it self-hosted, audit-first, with the bug-fix-agent loop closing all the way to a pull request.**

CUSTOMER PATTERN

EMPLOYMENT MARKETPLACE · LARGE JOB PORTAL

QA transitioned from Playwright

WHAT INSIGHTTESTBENCH DOES FOR THEM

Product owners now drive automated testing directly. The QA team's day shifted from authoring cases to reviewing AI output and triaging failures that need

OUTCOMES AFTER 12 MONTHS

2 wk → 2 d	30% → 8%	11	47 / 71
QA EFFORT · PER RELEASE	CI FLAKE · DISMISSED	REGRESSIONS CAUGHT PRE-RELEASE	BUG-FIX PRS MERGED (66%)

Move QA from authoring Playwright to operating an agent.

Pilot on one product surface. Product owner pastes a brief; the bench bootstraps the suite; schedule it nightly. The next regression comes in pre-classified with a suggested fix attached.

REQUEST PILOT →

AMS TRANSFORMATION

INSIGHTWORKER · L1 SUPPORT AGENTS

L1 support handled by agents. L2 engineers focus on the novel.

AMS teams spend most of their hours on triage, log analysis, and routine repeating tickets. InsightWorker agents take L1: triage every incoming ticket, classify it, resolve the routine ones autonomously, and hand the rest to L2 engineers with the log analysis, recent code changes, and a proposed fix already attached.

THE SHIFT

BEFORE

Engineers do the AMS investigation by hand.

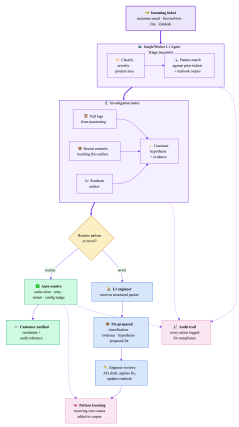
- L1 engineers triaging incoming tickets manually
- Each ticket = log search + recent commits + repro + diagnose
- Routine tickets eat 60-80% of AMS engineer hours
- Genuinely novel cases get only the leftover attention

AFTER · WITH INSIGHTWORKER L1

Agents do L1. Engineers handle the novel.

- Every ticket triaged, classified, and routed in seconds
- Routine tickets resolved autonomously with full audit trail
- L2 receives tickets pre-prepared: log analysis, code changes, fix proposal
- Recurring root-cause patterns recognized over time; runbook capacity grows

THE APPROACH



ALL THE AVAILABLE AI TOOLS — AND WHERE INSIGHTWORKER L1 FITS

ServiceNow Now Assist · Zendesk AI · Microsoft Copilot for Service · Custom RAG agents · Salesforce Agentforce · Lovable / v0 (prototypes) · **InsightWorker L1**

Vendor AI add-ons (ServiceNow, Zendesk, Microsoft, Salesforce) work well inside those ecosystems. Custom RAG agents work but each becomes a maintenance project. **InsightWorker L1 is the pick when you want self-hosted L1 / AMS automation that works across ticket systems, uses your own credentials, runs on your own infrastructure, and exposes every action in an open §15 audit log — no vendor lock-in.**

CUSTOMER PATTERN

BLOCKCHAIN · WORLD-LEADING FIRM

L1 transformation for the Application Management Services team

WHAT INSIGHTWORKER DOES FOR THEM

Deployed as the L1 layer. Incoming tickets triaged, classified, and either resolved autonomously or routed to L2 with pre-built investigations attached.

OUTCOMES

~70%	Hours → min	3×	100%
ROUTINE · AUTO-RESOLVED	TIME-TO-FIRST-RESPONSE	ENGINEER HOURS · NOVEL CASES	ACTIONS LOGGED FOR AUDIT

Make L1 the agent's job. Reserve your engineers for the novel cases.

Pilot on one product line's ticket queue. Mirror tickets to the L1 agent in parallel with existing routing; measure agreement and escalation quality. Cut over routine cases when the pattern's established.

REQUEST PILOT →